REMARKS

Applicant traverses the rejections, amends the specification to address a typographical issue, amends Figs. 1–3, amends claims 1 and 26, and cancels claims 3–5, 12 and 13. Claim 1 is amended to incorporate the subject matter of originally submitted claims 4 and 13 and to improve form. Claim 26 is amended to more particularly point out the claimed subject matter. No new matter is added. Applicant submits the enclosed Request for Continued Examination and respectfully requests reconsideration of the objections and rejections based upon the foregoing amendments and the following remarks.

Applicant acknowledges and appreciates the withdrawal of the objections to the specification and the acceptance of the Terminal Disclaimer.

The specification teaches an integrated document processing system. The system, as described, can receive an input stream corresponding to a number of predetermined computer applications and then drive an output device to present a visual image corresponding to that file without the use of that application. For example, the system can receive a Microsoft Word™ file, a PDF file, an HTML file, etc, and present a visual image corresponding to that file without the use of Microsoft Word™, Adobe Acrobat™, Internet Explorer™, or any other application external to the integrated system. The system accomplishes this task by converting the input stream into an internal representation that an integrated rendering engine can understand and utilize to drive an output device. The cited references fail to teach or describe such a system.

Amended Figures 1–3 Satisfy the Examiners Objections

Applicant amends Figs. 1–3 to add descriptive labels into the figures corresponding to the previously included reference numbers.

The References Fail To Teach or Suggest an Integrated Rendering Engine For Driving an Output Device Independent Of the Operation Of Other Applications

Amended claim 1 now includes the subject matter of originally filed claims 3 and 4, now cancelled, and recites an integrated digital document processing system. Amended claim 1 recites, in part, an adaptable front end for receiving an input stream generated by an external application, e.g., Microsoft Word™, Adobe Acrobat™, Microsoft's Internet Explorer™, etc. Specification at p. 4, ln. 30–p. 5, ln. 1. The input stream represents source data containing information representative of a visual image corresponding to a digital document. The

integrated system includes an interpreting module for generating an internal representation of the visual image. A rendering module, integrated into the system, drives an output device to display the visual image based on the internal representation independent of the operation of an external application.

Alam describes a document translation system that converts a document having an input format (e.g., PDF, RTF, HTML, XML, TIFF, etc.) into a second format referred to as the output format. Col. 1, ln. 60–Col. 2, ln. 11. The translation includes the conversion of the document from the input format into a plurality of intermediate format blocks. Col. 2, lns. 14–18. Fig. 7 of *Alam* indicates that the *Alam* intermediate format is output; however, *Alam* does not teach or suggest a system that processes the intermediate format *to drive an output device to present a visual image independent of any application external to the digital document processing system* as recited in claim 1. In contrast, the intermediate format is merely output into the output format. Col. 1, ln. 67–col. 2, ln. 1; Figures 2-6; Figures 11-13; Col 12, ln 55-63 To present the visual image encoded in the output format, additional processing is required. *Alam* fails to teach or suggest a device that includes an integrated rendering engine that can display this intermediate, source independent format. Yet this is the explicit subject matter recited in claim 1. Meltzer fails to bridge the gap.

Meltzer describes a system that receives XML documents in a first format at a network interface (step 400), converts the documents into a host format (step 403), such as a collection of JAVA objects. The system then converts the JAVA objects into an output format (step 406), such as XML. The process ends with the transmission of the output document to a destination through the network interface (step 407). The system fails to *drive an output device to present the digital document as a visual image* as recited in claim 1. To drive an output device to display a visual image, the system requires an XML viewer. *Meltzer* does not suggest that such a viewer is *integrated* into the system. Thus, an XML viewer *is an external application*, the operation of which to display a visual image is excluded in claim 1.

As neither *Alam* nor *Meltzer* describe an integrated system that includes a rendering engine for driving an output device to present a visual image independently of the operation of applications external to the integrated system, no combination of the references meets the limitations of claim 1. Applicant requests reconsideration and withdrawal of the §103 rejection of claim 1 on this basis. As claims 2, 6–11, and 14–34 depend (directly or indirectly) from claim

1 and further limitations thereto, Applicant respectfully requests reconsideration and withdrawal of the §103 rejections of these claims, as well.

The References Fail to Teach or Suggest the Features of the Interpreting Module Recited In Claim 1

In addition, neither *Alam* nor *Meltzer* teaches or suggests an interpreting module configured to generate an internal representation of a visual image, the features of which are explicitly recited in claim 1. The generation of the internal representation, as carried out by the interpreting module, includes the separation of the content of a document from the structures of the document. As the Examiner notes, *Alam* fails to teach or suggest an internal representation that separates the structure of document from the content of the document. *Meltzer*, too, fails to teach or suggest this subject matter.

In particular, the claimed interpreting module identifies *structure instances within a document*. The interpreting module stores an instance of a generic object, located from a library, to correspond to an identified structure instance. The interpreting module identifies and stores the content associated with the identified structure instance separately from the corresponding stored instance of a generic object .

The Action asserts that the combination of DTD files and JAVA Beans described in *Meltzer* constitutes separate storage of structure and content. Applicant disagrees. Neither the DTD files nor the JAVA objects in *Meltzer* satisfy the subject matter recited in claim 1 related to stored instances of generic objects instances and separately stored corresponding content. The only structure instances *Meltzer* describes as being identified within a document are XML logic structures. The identified XML logic structures are then translated into JAVA objects. Assuming, for the sake of argument, that the stored JAVA objects correspond to stored instances of generic objects, *Meltzer* does not describe an interpreting module that stores the content corresponding to a particular XML structure *apart* from the JAVA object corresponding to that XML structure; yet this is the explicit subject matter of claim 1. In fact, *Meltzer* teaches the exact opposite. *Meltzer* indicates that the JAVA objects "carry the data of the XML element[s]." Col. 26, lns. 26–28.

In columns 73–80, *Meltzer* provides sample code implementing this translation process. In translating a file, the *Meltzer* system sets the variable values of a JAVA object by parsing through a source document and finding XML tags described in a corresponding DTD file. Upon

finding such a tag, the system calls functions defined in the JAVA Bean corresponding to the DTD file (see cols. 34–71). For example, with respect to a "party name," the *Meltzer* system uses a setter function called "partyNameFromXML" (see bottom col. 73) found in the JAVA Bean at col. 51. This function sets the value of a variable this.mPartyName to the party name listed in the XML file.

A specific example of an XML document containing the "party name" structure is given at col 79-80, with the text string "IBM" set as the content of the structure. (See line 4 of the example XML document). In translating this XML document, the Meltzer system employs the Java class "partyNameFromXML" to extract the contents (i.e. the string "IBM") from the XML structure and sets this same content into the Java object that constitutes the intermediate format. The end result of the translation is a JAVA object that includes a number of variables (corresponding to the XML logic structures) having values that are set equal to the content of the XML structure. In the cited example, the Java object will contain the statement:

party.name = IBM

Clearly this representation does not store the content (IBM) apart from the stored instance of the identified structure (party name), as recited in amended Claim 1.

As neither *Alam* nor *Meltzer* describe an interpreting module configured to generate an internal representation of a visual image by identifying a plurality of structure instances within a document, storing an identified structure instance as an instance of a generic object, and storing content associated with the structure instance apart from the generic object instance corresponding to the identified structure instance, no combination of the references can result in a system that meets the limitations of claim 1. Applicant requests reconsideration and withdrawal of the §103 rejection of claim 1 on this basis. As claims 2, 6–11, and 14–34 depend (directly or indirectly) from claim 1 and further limitations thereto, Applicant respectfully requests reconsideration and withdrawal of the §103 rejections of these claims, as well.

The Combination of the Cited References Is Improper

As previously argued, *Meltzer* and *Alam* cannot properly be combined to support a prima facie obviousness rejection because such a combination would change the principle of operation of either of the references in contravention of MPEP § 2143. In particular, *Alam* is free from the constraints of *Meltzer*'s predefined input document structures, and any combination of the two would undermine *Alam*'s achieved flexibility, which is the whole point of the *Alam* system. In

response to Applicant's remarks in this regard, the Action asserts that the "combination of references lies within the modification of the internal representation of *Alam* with that of *Meltzer*." However, the combination asserted by the Action explicitly includes the use of DTD files. See p. 3. DTD files are the very reason for the limited flexibility of *Meltzer* as DTD files predefine the structure of a document. Therefore, the host format of *Meltzer* cannot be incorporated into *Alam* without violating a principle of operation of *Alam*, its flexibility with respect to input document structure.

In light of the above, Applicant respectfully requests reconsideration and withdrawal of the §103 rejections of claim 1. As claims 2, 6–11, and 14–34 depend (directly or indirectly) from claim 1 and further limitations thereto, Applicant respectfully requests reconsideration and withdrawal of the §103 rejections of these claims, as well.

CONCLUSION

In view of the above, each of the presently pending claims in this application is believed to be in immediate condition for allowance. Accordingly, the Examiner is respectfully requested to pass this application to issue.

Applicant believes no fee is due with this response other than as reflected on the enclosed Fee Transmittal. However, if a fee is due, please charge our Deposit Account No. 18-1945, under Order No. PGLD-P01-001 from which the undersigned is authorized to draw.

Dated: November 16, 2004

Respectfully submitted,

By _____

Edward A. Gordon
Registration No.: 54,130
ROPES & GRAY LLP
One International Place
Boston, Massachusetts 02110-2624
(617) 951-7000
(617) 951-7050 (Fax)
Attorneys/Agents For Applicant

Attachments

AMENDMENTS TO THE DRAWINGS

The attached sheets of drawings includes changes to Figures 1–3. The Figures are amended to add descriptive labels to supplement the reference numerals.

Attachment: Replacement sheets